Amendments to Specification

On page 5, lines 9-13, please replace the existing paragraphs with the following substitute paragraphs:

~~FIGURES 4A-C~~ FIGURES 4(A)-(C) are data representation diagrams showing anchor points within cluster spines.

FIGURE 5 is a data representation diagram showing, by way of example, a view of a thematically-related cluster spine grafted onto the cluster spine of ~~FIGURE 4~~ FIGURE 3.

On page 5, lines 20-22, please replace the existing paragraph with the following substitute paragraph:

FIGURE 8 is a data representation diagram showing, by way of example, a view of a thematically-related cluster spine grafted onto an end-point cluster of the cluster spine of ~~FIGURE 4~~ FIGURE 3.

On page 5, lines 30-31, please replace the existing paragraph with the following substitute paragraph:

~~FIGURE 12 is~~ FIGURES 12(A)-(B) are a routine for placing clusters for use in the method of FIGURE 8.

On page 8 lines 20 through page 9, line 2, please replace the existing paragraphs with the following substitute paragraphs:

In the described embodiment, cluster size equals the number of concepts contained in the cluster. The cluster spine [[41]] 42 is built by identifying those clusters 44-46 sharing a common theme. A theme combines two or more concepts 47, which each group terms or phrases (not shown) with common semantic meanings. Terms and phrases are dynamically extracted from a document collection through latent concept evaluation. During cluster spine creation, those clusters 44-46 having available anchor points [[48]] within each

cluster spine [[~~41~~]] 42 are identified for use in grafting other cluster spines sharing thematically-related concepts, as further described below with reference to FIGURE 5.

The cluster spine [[~~41~~]] 42 is placed into a visual display area to generate a two-dimensional spatial arrangement. To represent data inter-relatedness, the clusters 44-46 in each cluster spine [[~~41~~]] 42 are placed along a vector [[44]] 43 arranged in decreasing cluster size, although other line shapes and cluster orderings can be used.

On page 9 lines 3-8, please replace the existing paragraph with the following substitute paragraph:

~~FIGURES 4A-C~~ FIGURES 4(A)-(C) are data representation diagrams 50, 60, 65 respectively showing anchor points within cluster spines 51, 61, 66. A cluster having at least one open edge constitutes an anchor point. Referring first to ~~FIGURE 4(A),~~ FIGURE 4(A), a largest endpoint cluster 52 of a cluster spine 51 functions as an anchor point along each open edge 55a-e. The endpoint cluster 52 contains the largest number of concepts.

On page 9 lines 21-26, please replace the existing paragraph with the following substitute paragraph:

Referring next to ~~FIGURE 4(B),~~ FIGURE 4(B), a smallest endpoint cluster 62 of a cluster spine 61 also functions as an anchor point along each open edge. The endpoint cluster 62 contains the fewest number of concepts. The clusters in the cluster spine 61 are arranged in order of decreasing cluster size. An open edge is formed by projecting vectors 64a-c outward from the center 63 of the endpoint cluster 62, preferably at normalized angles.

On page 10 lines 1-10, please replace the existing paragraph with the following substitute paragraph:

Referring finally to ~~FIGURE 4C,~~ FIGURE 4(C), a midpoint cluster 67 of a

cluster spine 61 functions as an anchor point for a cluster spine 66 along each open edge. The midpoint cluster 67 is located intermediate to the clusters in the cluster spine 66 and defines an anchor point along each open edge. An open edge is formed by projecting vectors 69a-b outward from the center 68 of the midpoint cluster 67, preferably at normalized angles. Unlike endpoint clusters 52, 62 the midpoint cluster 67 can only serve as an anchor point along tangential vectors non-coincident to the vector forming the cluster spine 66. Accordingly, endpoint clusters 52,62 include one additional open edge serving as a coincident anchor point.

On page 10, lines 16-23, please replace the existing paragraph with the following substitute paragraph:

FIGURE 5 is a data representation diagram 70 showing, by way of example, a view 71 of a thematically-related cluster spine [[~~75~~]] 72 grafted onto the cluster spine 42 of FIGURE 3. Each cluster in the cluster spine [[~~75~~]] 72, including endpoint cluster 74 and midpoint clusters 75, share concepts in common with the midpoint cluster 76 of the cluster spine 42. Accordingly, the cluster spine 72 is "grafted" onto the cluster spine 42 at an open edge of an available anchor point on midpoint cluster 76. The combined grafted clusters form a cluster grouping or "grouper" of clusters sharing related or similar themes.

On page 10 lines 24 through page 11, line 2, please replace the existing paragraph with the following substitute paragraph:

FIGURE 6 is a data representation diagram 80 showing, by way of example, a view 81 of singleton clusters 86 and further thematically-related cluster spines 82 and 84 grafted onto the cluster spine 42 of FIGURE 3. The clusters in the cluster spines 82 and 84 share concepts in common with the clusters of cluster spine 42 and are grafted onto the cluster spine 82 at open edges of available anchor points. Slight overlap 87 between grafted clusters is allowed. In the described embodiment, no more than 20% of a cluster can be covered by

overlap. The singleton clusters 86, however, do not thematically relate to the clusters in cluster spines [[~~62~~]] 42, 72, 82, 84 and are therefore grouped as individual clusters in non-relational placements.

On page 11, lines 21-23, please replace the existing paragraph with the following substitute paragraph:

FIGURE 8 is a data representation diagram 110 showing, by way of example, a view 111 of a thematically-related cluster spine grafted onto an end-point cluster of the cluster spine of FIGURE 3.

On page 12, lines 1-5, please replace the existing paragraph with the following substitute paragraph:

FIGURE 9 is a flow diagram 120 showing a method for arranging concept clusters in thematic relationships in a two-dimensional visual display space [[~~120,~~]] in accordance with the present invention. The method presents arbitrarily dimensioned concept data visualized in a two-dimensional visual display space in a manner that preserves independent data relationships between clusters.

On page 12, lines 22-28, please replace the existing paragraph with the following substitute paragraph:

Each cluster is iteratively sized in a processing loop (blocks 131-133) as follows. For each cluster processed in the processing loop (block 131), the cluster size is set to equal the number of concepts contained in the cluster (block 132). Iterative processing continues (block 133) for each remaining cluster. The groupers are then placed into the visual display space (block 134), as further described below with reference to FIGURE 13. Finally, the placed groupers are displayed (block 135), after which the routine terminates.

On page 13, line 3-page 14, line 6, please replace the existing paragraphs with the following substitute paragraphs:

The sublists are built by iteratively processing each shared concept in an outer processing loop (blocks 141-150) as follows. For each new shared concept processed in the outer processing loop (block 141), a sublist of clusters belonging to the shared concept is built (block 142). A cluster center represents a seed value originating from the shared concept. A seed value typically consists of the core set of concepts, preferably including one or more concepts, which form the basis of the current sublist. Thereafter, each of the clusters is iteratively processed in an inner processing loop (blocks 143-149) to determine sublist membership, as follows.

For each cluster processed in the inner processing loop (block 143), if the cluster does not belong to the current sublist (block 144), that is, the cluster does not share the common concept, the cluster is skipped (block 149). Otherwise, if the cluster has not already been placed in another sublist (block 145), the cluster is added to the current sublist (block 146). Otherwise, if the cluster has been placed (block 145) and has an open edge (block 147), the cluster is marked as a anchor point (block 148). Iterative processing of each cluster (block 149) and shared concept (block 150) continues, after which the routine returns.

~~FIGURE 12 is~~ FIGURES 12(A)-(B) are a routine for placing clusters 160 for use in the method of FIGURE 9. The purpose of this routine is to form cluster groupings or "groupers" of grafted cluster spines.

Each sublist of placeable clusters is iteratively processed in an outer processing loop (blocks 161-175), as follows. For each sublist processed in the outer processing loop (block 161), if the sublist includes an anchor point (block 162), the anchor point is selected (block 165). Otherwise, a new grouper is started (block 163) and the first cluster in the sublist is selected as the anchor point and removed from the sublist (block 164). Each cluster in the sublist is then iteratively processed in an inner processing loop (blocks 166-173), as follows.

For each cluster processed in the inner processing loop (block 166), the radius of the cluster is determined (block 167) and the routine attempts to place the cluster along the open vectors emanating from the anchor point (block 168).

The radius is needed to ensure that the placed clusters do not overlap. If the cluster was not successfully placed (block 169), the cluster is skipped and processed during a further iteration (block 175). Otherwise, if the cluster is successfully placed (block 169) and is also designated as an anchor point (block 170), the angle of the anchor point is set (block 171), as further described below with reference to ~~FIGURE 12~~ FIGURES 12(A)-(B). The cluster is then placed in the vector (block 172). Processing continues with the next cluster (block 173).

On page 14, line 28-page 15, line 15, please replace the existing paragraphs with the following substitute paragraphs:

Each of the groupers is iteratively processed in a processing loop (blocks 191-197), as follows. For each grouper processed in the processing loop (block 191), if the grouper comprises a singleton cluster (block 192), the grouper is skipped (block 197). Otherwise, if the grouper is the first grouper selected (block 193), the grouper is centered at the origin of the visual display space (block 194). Otherwise, the angle of the grouper and radius from the center of the display are incremented by the size of the grouper, plus extra space to account for the radius of the end-point cluster at which the cluster is grafted (block 195) until the grouper can be placed without substantially overlapping any previously-placed grouper. Slight overlap within 20° between clusters is allowed. A grouper is added to the display space (block 196). Iterative processing continues with the next grouper (block 197). Finally, all singleton groupers are placed in the display space (block 198). In the described embodiment, the singleton groupers are placed arbitrarily in the upper left-hand corner, although other placements of singleton groupers are possible, as would be recognized by one skilled in the art. The routine then returns.

Although the foregoing method 120 of ~~FIGURE 8~~ FIGURE 9 has been described with reference to circular clusters, one skilled in the art would recognize that the operations can be equally applied to non-circular clusters forming closed convex volumes.